

# Manual de Usuario

## 1. Requisitos Previos

- **Cliente API:** Se recomienda utilizar Postman para realizar pruebas y consumir la API.
  - **Entorno de Ejecución:** Asegúrese de que la API se esté ejecutando correctamente. Por ejemplo, inicie el servidor Flask (por defecto en `http://127.0.0.1:5000` ).
  - **Archivos de Colección Postman:** Se han creado colecciones para cada módulo (Órdenes, Pagos, Productos y Usuarios) que facilitan la prueba de cada endpoint.
- 

## 2. Importar Colecciones en Postman

1. **Abrir Postman:** Inicie la aplicación Postman en su equipo.
  2. **Importar Colección:**
    - Haga clic en el botón **Import** en la esquina superior izquierda.
    - Seleccione la opción **Upload Files** y elija el archivo de la colección correspondiente:
      - `Ordenes.postman_collection.json`
      - `Pagos.postman_collection.json`
      - `Productos.postman_collection.json`
      - `Usuarios.postman_collection.json`
    - Una vez importado, las colecciones aparecerán en la sección **Collections** de Postman.
- 

## 3. Configuración del Entorno en Postman

Aunque el API se ejecuta en `http://127.0.0.1:5000` por defecto, puede configurar variables de entorno en Postman para centralizar la URL base:

1. **Crear un Nuevo Entorno:**

- Haga clic en el ícono de engranaje en la esquina superior derecha y seleccione **Manage Environments**.
- Cree un nuevo entorno (por ejemplo, "Local API").

## 2. Agregar Variable `base_url` :

- Defina la variable `base_url` con el valor `http://127.0.0.1:5000` .

## 3. Asignar el Entorno:

- Seleccione el entorno "Local API" en el desplegable de entornos en la parte superior derecha.

*Nota:* Algunas colecciones ya incluyen la URL completa, pero es útil para centralizar configuraciones en caso de cambios.

---

# 4. Consumo de Endpoints

Cada colección contiene un conjunto de ejemplos de solicitudes que cubren las operaciones principales de cada módulo. A continuación se explica brevemente cómo consumir cada uno de ellos.

## 4.1 Módulo de Órdenes ( `/api/orders` )

- **Crear Orden de Compra:**

- **Método:** `POST`
- **URL:** `{{base_url}}/api/orders`
- **Body (JSON):**

```
{
  "userId": 1,
  "items": [
    { "productId": 6, "quantity": 2 },
    { "productId": 5, "quantity": 1 }
  ],
  "shippingAddress": "Avenida Siempre Viva 742, Ciudad Norte",
  "paymentMethodId": 1
}
```

- **Uso:** Enviar esta solicitud para crear una nueva orden de compra. La respuesta incluirá el identificador de la orden, monto total, y estado de la orden.
- **Listar Órdenes:**
  - **Método:** GET
  - **URL:** `{{base_url}}/api/orders`
  - **Uso:** Realizar esta solicitud para obtener un listado de todas las órdenes. Se puede incluir parámetros de consulta si fuera necesario.
- **Detalle de Orden:**
  - **Método:** GET
  - **URL:** `{{base_url}}/api/orders/<id>` (reemplazar `<id>` por el identificador de la orden deseada)
  - **Uso:** Esta solicitud retorna la información completa de una orden específica, incluyendo los ítems y totales.

## 4.2 Módulo de Pagos ( `/api/payments` )

- **Registrar Pago:**
  - **Método:** POST
  - **URL:** `{{base_url}}/api/payments`
  - **Body (JSON):**

```
{
  "orderId": 7,
  "amount": 800.00,
  "paymentMethodId": 1
}
```
  - **Uso:** Enviar esta solicitud para registrar un pago para una orden existente. La respuesta indicará el estado del pago (aprobado o pendiente), el total pagado y el monto restante.
- **Consultar Pagos:**
  - **Método:** GET

- **URL:** `{{base_url}}/api/payments`
- **Uso:** Utilice esta solicitud para listar los pagos registrados. Puede incluir parámetros de consulta, como `orderId`, `paymentMethodId`, `dateFrom` y `dateTo` para filtrar resultados.

## 4.3 Módulo de Productos ( `/api/products` )

- **Listar Productos:**

- **Método:** `GET`
- **URL:** `{{base_url}}/api/products`
- **Uso:** Obtenga un listado de productos activos con su información básica.

- **Detalle de Producto:**

- **Método:** `GET`
- **URL:** `{{base_url}}/api/products/<id>`
- **Uso:** Retorna la información detallada de un producto específico (reemplazar `<id>` ).

- **Crear Producto:**

- **Método:** `POST`
- **URL:** `{{base_url}}/api/products`
- **Body (JSON):**

```
{
  "name": "Monitor Gamer 27\"",
  "description": "Monitor curvo de 27 pulgadas, 165Hz, 1ms de respuesta",
  "price": 350.00,
  "stock": 15,
  "category": "Electrónicos",
  "location": "Sede Central"
}
```

- **Uso:** Permite agregar un nuevo producto al catálogo.

- **Actualizar Producto:**

- **Método:** PUT
- **URL:** {{base\_url}}/api/products/<id>
- **Body (JSON):**

```
{
  "price": 5000.00,
  "stock": 25,
  "location": "Sede Norte"
}
```

- **Uso:** Modifica la información de un producto existente.

- **Eliminar Producto:**

- **Método:** DELETE
- **URL:** {{base\_url}}/api/products/<id>
- **Uso:** Marca un producto como inactivo en el sistema.

## 4.4 Módulo de Usuarios ( /api/users )

- **Crear Usuario:**

- **Método:** POST
- **URL:** {{base\_url}}/api/users
- **Body (JSON):**

```
{
  "username": "jdoe123",
  "email": "jdoe@example.com",
  "password": "securePass1",
  "phone": "+1234567890",
  "national_document": "A12345678",
  "name": "John",
  "lastname": "Doe"
}
```

- **Uso:** Registra un nuevo usuario en el sistema.

- **Crear Dirección:**

- **Método:** POST

- **URL:** `{{base_url}}/api/users/add_address`

- **Body (JSON):**

```
{
  "username": "jdoe123",
  "address": "123 Main Street, City, Country"
}
```

- **Uso:** Asocia una dirección al usuario especificado.

- **Inicio de Sesión:**

- **Método:** `POST`

- **URL:** `{{base_url}}/api/users/login`

- **Body (JSON):**

```
{
  "username": "jdoe123",
  "password": "securePass1"
}
```

- **Uso:** Autentica al usuario y devuelve un identificador de sesión.

- **Obtener Usuario:**

- **Método:** `GET`

- **URL:** `{{base_url}}/api/users/<id>`

- **Uso:** Recupera la información del usuario basado en su ID.

- **Actualizar Usuario:**

- **Método:** `PUT`

- **URL:** `{{base_url}}/api/users/<id>`

- **Uso:** Actualiza datos del usuario (por ejemplo, teléfono y/o email).

- **Eliminar Usuario:**

- **Método:** `DELETE`

- **URL:** `{{base_url}}/api/users/<id>`

- **Uso:** Marca el usuario como inactivo en el sistema.
  - **Listar Métodos de Pago del Usuario:**
    - **Método:** GET
    - **URL:** {{base\_url}}/api/users/payment\_methods
    - **Uso:** Recupera los métodos de pago asociados al usuario.
- 

## 5. Ejecución de Pruebas

### 1. Importe las colecciones:

Siga los pasos descritos en la sección 2 para importar las colecciones a Postman.

### 2. Configure el entorno:

Asegúrese de que la variable

`base_url` esté configurada correctamente (por ejemplo, `http://127.0.0.1:5000` ).

### 3. Ejecute las solicitudes:

- Seleccione una solicitud de la colección.
- Haga clic en el botón **Send** para enviar la petición.
- Verifique la respuesta en el panel de salida.

### 4. Modificar Parámetros:

Puede editar el body o los parámetros de la URL para probar distintos escenarios y validar la robustez de la API.

---